

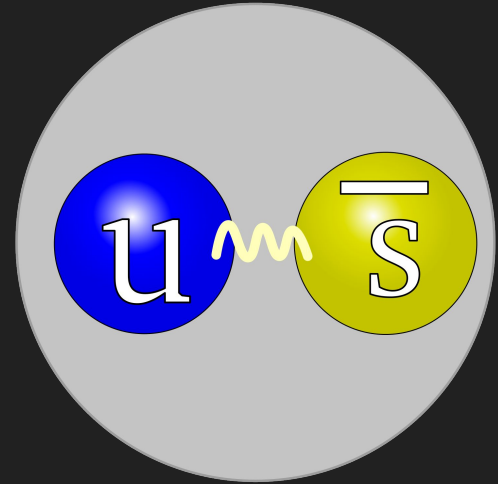


NA62 Triggering

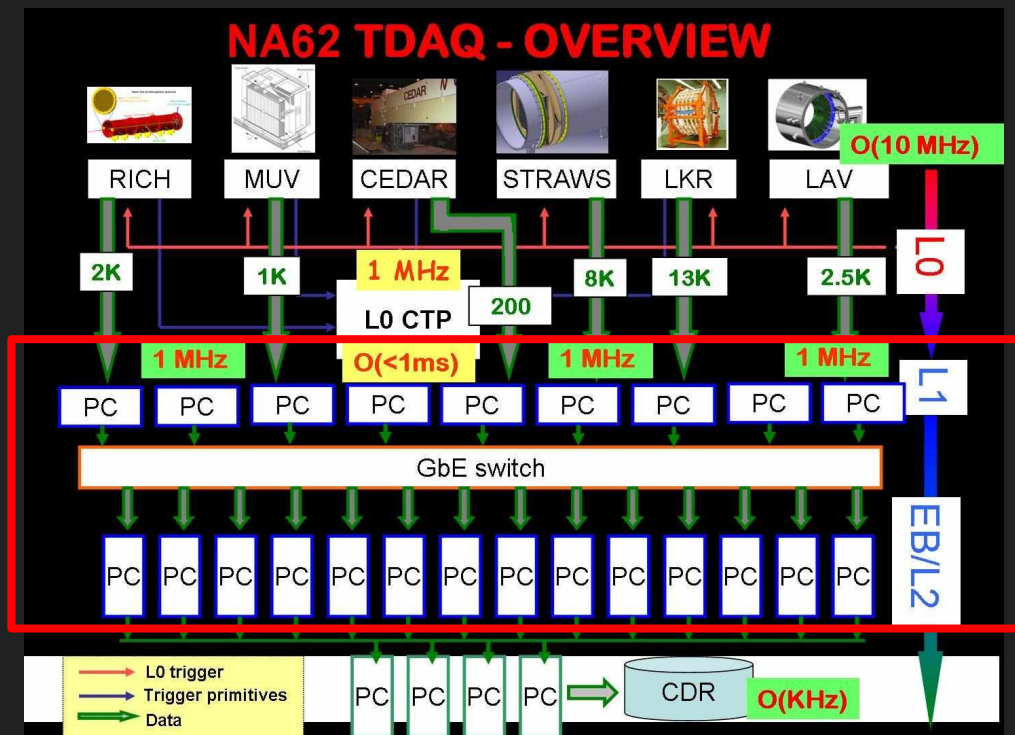
Adam Pearson

What is NA62 (Kaon Factory)

- “NA62 initially proposed to measure the very rare kaon decay $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ at the CERN SPS to extract a 10% measurement of the CKM” (Cabibbo–Kobayashi–Maskawa/quark mixing matrix) “parameter $|V_{td}|$.” (Squaring this gives the probability of a top quark transitioning into a down quark, currently the best measurements give this to be about .00867)
- Uses “400 GeV/c protons from the SPS”

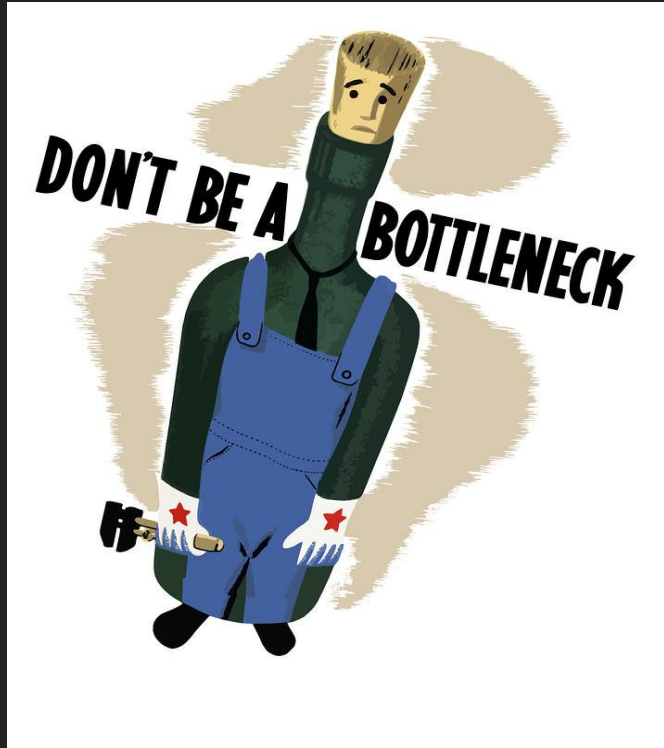


TDAQ



- Many detectors/layers of processing ultimately give bursts of data fragments to the L1 Triggering
- Triggering must be done extremely quickly (currently limited by time before next burst)

What's Wrong with This?



- Currently there is one copy of the data that is tossed around from process to process
- If any process crashes or slows down for any reason, it causes the entire chain of processes to crash or slowdown... textbook bottleneck
- If any data isn't processed fully before the next burst comes in, it is thrown out
 - Event IDs are reused in each burst, so Event 1 from Burst 1 could/would conflict with Event 1 from Burst 2... how to deal with this other than just throwing out the old stuff?

Initial Solution?

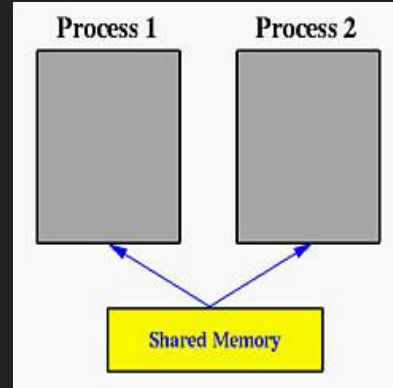
- Decouple the processes, i.e. allow them to all act independently of each other!

... But then what happens to that one copy of data being tossed between processes, what happens if two processes want to change the same data or if one process dies while working on a piece of data...???? This seems awfully dangerous from a memory management point of view!



How Ever Could These Problems Be Solved??

- MANAGED SHARED MEMORY
- There are several ways to do this, but a nice little C library from Boost (a collection of nifty/experimental C libraries) called Interprocess offers a very clean solution
- Implementing this in the current code is now the difficulty...
- The only seeming downside is more memory overhead... but it's persistent!



- As for throwing old events out... why not label each one with a burst ID as well (this will only delay the eventual need to throw out, but by a large amount)
- Also, speeding up current processing will reduce the need to throw old events out even more

Main Sources

<https://na62.web.cern.ch/NA62/>

https://en.wikipedia.org/wiki/Cabibbo%E2%80%93Kobayashi%E2%80%93Maskawa_matrix